
This is the **published version** of the bachelor thesis:

Susín Palacios, Martín; Romero Tris, Cristina, dir. Dog to house : sistema per trobar geogràficament al nostre animal de companyia. 2021. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/248523>

under the terms of the  license

Dog to House: Sistema per trobar geogràficament al nostre animal de companyia

Martin Susin Palacios

Resum– Sistema basat en Arduino, complementat amb una petita aplicació Android, per avisar-nos de la posició geogràfica del nostre animal de companyia, si ens trobem massa lluny d'ell. L'aplicació Android permetrà veure un mapa amb la localització de l'animal, les alertes i el sistema per desactivar-les, un cop introduïm el codi que es mostri a la pantalla. El sistema funcionarà de la següent manera, el mòbil estarà connectat al mòdul Arduino per via Bluetooth i s'haurà de fer via l'aplicació que es desenvoluparà. En el cas que el gos es trobi massa lluny, el sistema Arduino farà diferents funcions, per una part enviarà una localització, que se sabrà gràcies al GSM, al dispositiu mòbil on es veurà la posició exacta del gos, aquesta localització s'enviarà cada 5 minuts per poder saber on està en tot moment.

Paraules clau– Arduino, Geolocalització, Sistema d'alarmes, Animals de companyia

Abstract– Arduino-based system, complemented by a small Android application, to warn us of the geographical position of our pet, if we are too far from it. The Android application will allow you to see a map with the location of the animal, the alerts and the system to deactivate them, once we enter the code displayed on the screen. The system will work as follows, the mobile will be connected to the arduino module via Bluetooth and the application to be developed will have to be done. In the event that the dog is too far away, the Arduino system will perform different functions, on the one hand it will send a location, which will be known thanks to the GSM, the mobile device where you will see the exact position of the dog, this location will be sent every 5 minutes to be able to know the bone is at all times.

Keywords– Arduino, Geolocation, Alarm system, Pets



1 Introducció

1.1 Context del treball

DES de fa molts anys es poden trobar pel carrer cartells d'animals perduts, ja siguin gossos i/o gats que han desaparegut per diferents circumstàncies. Normalment aquests animals no es poden trobar i això succeeix perquè els animals no porten cap eina i/o accessori per poder trobar-los. Els animals que tenen més sort acaben en un centre d'acollida on els poden retornar a la seva família.

El sistema Dog To House vol reduir significativament el percentatge d'animals que s'escapen i no es poden trobar. Com a tecnologies que es volen implementar per a la creació del sistema de Dog To House, es pot trobar com a principal, la geocalització via per un xip GSM (Sistema global per les comunicacions mòbils) que d'aquí es pot treure la latitud i longitud per poder saber la posició. A més per poder utilitzar el xip GSM, s'utilitzarà Arduino, que avui dia està molt estandarditzat i és fàcil d'utilitzar, per

poder programar-lo s'utilitzarà el llenguatge de programació C++, s'ha decidit utilitzar aquest sobretot perquè s'ha estat utilitzant des que vaig començar la universitat. Per poder visualitzar aquestes dades s'utilitzarà una aplicació creada amb llenguatge JAVA, decidit aquest perquè s'ha estat utilitzant, per poder desenvolupar aplicacions android, durant el grau d'Enginyeria.

Tota aquesta idea ha vingut a causa que durant tots aquests anys he observat per molts llocs d'informació moltes notícies sobre animals perduts i molta gent parlant sobre aquests casos. Aquests fets van fer que comences a pensar sobre alguna idea per poder fer que tots els casos baixin el més possible. D'aquesta manera va aparèixer la idea de Dog To House.

Els objectius d'aquest sistema que es volen realitzar i arribar a fer en aquest treball es veuen en la següent taula:

Objectius	Descripció	Prioritat
O.1	Reduir la taxa d'animals que es perden a causa de que s'escapen.	Crític
O.2	Reduir la mortalitat dels animals "abandonats".	Crític
O.3	Augmentar la confiança de la persona que el seu animal estarà bé.	Prioritari

Per poder complir aquests objectius s'hauràn de seguir les següents fases:

Fases	Descripció
Base	Crear un sistema virtual que tingui per una part una virtualització d'una placa d'Arduino que s'encarregarà d'enviar la posició, per l'altra part un emulador Android que la imprimirà en format de mapa.
Final	Fer un estudi de mercat per veure si la idea de fer la implementació en Arduino és factible o s'hauria de buscar uns altres tipus de placa.
Ampliació	Passar el sistema virtual a un sistema físic, que contindrà una placa d'Arduino connectada per Bluetooth al mòbil, amb la implementació de notificacions per part de l'aplicació.
Ampliació	Implementar que es notifiqui a l'aplicació cada 5 minuts amb la posició exacta de l'animal.

En les següents seccions es veurà els passos que es portaran a terme i com es faran aquests de forma més detallada.

1.2 Estat de l'art

Actualment en el mercat hi ha una gamma de dispositius de la marca Tractive, que tenen com a finalitat la mateixa que el sistema Dog To House. Aquest sistema permet veure al teu animal de companyia per un mapa i saber quan s'ha t'ha escapat. Per poder utilitzar aquest sistema s'ha de pagar el dispositiu, d'entre trenta a setanta euros més una subscripció anual de seixanta euros.

Aquí és on entra el sistema Dog To House que com s'ha dit implementa una part d'aquest sistema i s'hauria de pagar el dispositiu amb una subscripció mensual/trimestral/anual reduïda del preu original del dispositiu Tractive.

2 Metodologia

En aquest apartat s'explicaran quins i com seran els passos a seguir per poder fer el sistema. Aquests passos estan explicats de forma detallada en les subseccions (2.1, 2.2, 2.3 i 2.4).

2.1 Pressa de requisits

En aquest apartat es descriuen els requisits que tindrà el projecte, així com un petit estudi de la situació actual d'aquest i la selecció de solucions i conclusions que poden trobar. Tot això servirà per tenir una base per treballar el projecte i tenir una base per començar a desenvolupar aquest.

2.1.1 Requisits del sistema

El sistema està compost per diverses parts tècniques com ara una aplicació mòbil, un sistema Arduino que portarà integrat el xip GSM i l'antena Bluetooth per poder saber quan s'escapa l'animal.

Per poder extreure els requisits s'ha fet un estudi per GoogleForms [4] d'on s'ha extret molta informació per poder definir de forma clara els requisits que hauria de tenir el sistema Dog To House.

2.1.2 Requisits funcionals

En la següent taula es podrà observar els principals requisits funcionals del projecte, amb una identificació, una petita descripció.

Identificació	Descripció
R.1	Fer una vinculació via Bluetooth entre la placa i el mòbil.
R.2	Realització d'un registre i inici de sessió per poder entrar en l'aplicació.
R.3	Poder visualitzar en un mapa la posició del teu animal de companyia.
R.4	Poder visualitzar la direcció del carrer on està el teu animal de companyia.
R.5	Poder desactivar el sistema una vegada s'ha trobat animal.
R.6	Poder desactivar el sistema per un error d'aquest.

2.1.3 Requisits no funcionals

En la següent taula es podrà observar els principals requisits no funcionals del projecte, amb una identificació, una petita descripció d'aquest i l'importància.

Identificació	Descripció	Importància
RN.1	La placa d'Arduino ha de tenir unes mides mitjanes.	Essencial
RN.2	L'aplicació mòbil ha de ser intuïtiva per a totes les edats.	Essencial
RN.3	Implementar el sistema de registre i inici de sessió via API Google.	Opcional

2.1.4 Restriccions del sistema

- Tenim restriccions a l'hora de treballar amb les dades dels tutors legals. Hem de tenir en compte la llei de protecció de dades.
- Hem de procurar que el temps de resposta del sistema actuï en el menor temps possible.
- Una restricció important és el senyal de cobertura que tingui el dispositiu per poder emetre el senyal d'alarma si s'escau.

2.1.5 Requisits futurs

A causa de voler millorar el sistema, cada vegada que es tingui una versió prototip es faran entrevistes amb persones que tinguin animals de companyia, per tal d'arribar al públic més gran possible i arribar a la majoria de persones, a més que així possibles requisits que hi ha ara es poden modificar o millorar per a entregues futures.

2.2 Disseny del sistema

Per fer la part de disseny es desenvoluparà una serie de diagrames per poder entendre de forma clara tot el sistema, entre aquests diagrames tindrem:

- Diagrama de casos d'ús
- Diagrama de classe

Per tenir una idea del que es vol desenvolupar s'utilitzarà com a referència aquest esquema:

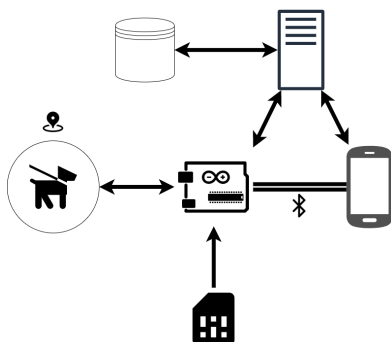


Fig. 1: Esquema físic del Sistema

En aquesta figura podem diferenciar tres parts que es desenvoluparan per separat:

- La placa d'Arduino que portarà un xip GSM i una antena Bluetooth per poder saber la posició exacta del mòbil.
- Una aplicació Android que serà on es connectarà la persona per poder vincular el seu mòbil i així rebre notificacions d'on és el seu animal.
- Un servidor que s'encarregarà de guardar les dades que arribaran des de la placa i també serà des d'on l'aplicació agafarà les coordenades per visualitzar-les per pantalla. Aquesta part s'ha afegit nova, ja que sense aquesta, no hi ha possibilitat directa que el xip GSM enviï la informació a l'aplicació, l'únic que permet és enviar SMS, aquesta informació s'ha extret de [7].

També s'hauria de dissenyar on es podria posar la placa d'Arduino en l'animal, en primera instància la idea, seria que estigués en el collar de l'animal.

2.2.1 Diagrama de casos d'ús

En aquesta part es podrà veure el diagrama que està especialitzat en casos d'ús amb una breu explicació de cada un d'ells.

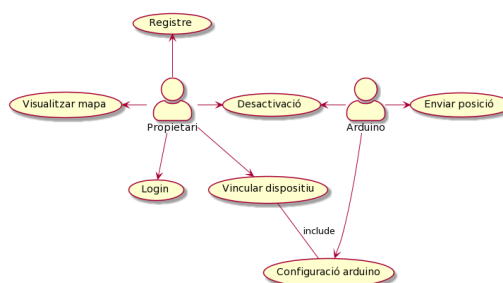


Fig. 2: Diagrama de casos d'ús

En les següents taules es podran veure les explicacions de cada un dels casos d'ús que hi ha, amb una explicació per aclarir que fa cada part.

Propietat	Descripció
ID. Cas d'ús	CU-01
Títol	Login
Precondicions	No ha d'haver-hi cap sessió iniciada. L'usuari ha d'estar iniciat.
Postcondicions	No es pot realitzar el login una vegada ja s'ha fet correctament.
Escenari principal	<ol style="list-style-type: none"> 1. El sistema demana a credencials a l'usuari. <ol style="list-style-type: none"> (a) nom (b) contrasenya 2. L'usuari introdueix al sistema les credencials. 3. L'usuari clica el botó d'Inici de sessió. 4. El sistema ensenya una pantalla de càrrega.

Propietat	Descripció
ID. Cas d'ús	CU-03
Títol	Visualitzar mapa
Precondicions	L'usuari ha d'haver donat permisos de geocalització. L'usuari ha d'haver-hi vinculat el dispositiu.
Postcondicions	No es pot sortir fins que és tingui a prop de l'animal.
Escenari principal	<ol style="list-style-type: none"> 1. El sistema agafa les coordenades del servidor. 2. El sistema carrega un mapa sobre la posició de l'animal. 3. El sistema carrega un marca d'on està l'animal. 4. El sistema carrega la posició on està l'usuari actualment. 5. El sistema carrega la marca de l'usuari. 6. El sistema carrega un possible camí fins a l'animal.

Propietat	Descripció
ID. Cas d'ús	CU-02
Títol	Registre
Precondicions	No ha d'haver-hi una sessió iniciada. L'usuari no s'ha registrat.
Postcondicions	L'usuari podrà fer Login una vegada registrat.
Escenari principal	<ol style="list-style-type: none"> 1. El sistema demana a la diferent informació a l'usuari. <ol style="list-style-type: none"> (a) Nom (b) Cognom (c) contrasenya (d) repetir contrasenya (e) nombre d'animals 2. L'usuari introdueix al sistema la informació. 3. El sistema verifica que la informació es valida. 4. El sistema verifica que la informació te'l format correcte. 5. El sistema avisa a l'usuari que s'hi ha registrat correctament. 6. El sistema retorna a la pantalla d'inici de sessió/registre.

Propietat	Descripció
ID. Cas d'ús	CU-04
Títol	Vincular dispositiu
Precondicions	L'usuari ha de tenir la sessió iniciada.
Postcondicions	Solament es podrà observar l'animal amb el dispositiu vinculat.
Escenari principal	<ol style="list-style-type: none"> 1. L'usuari pitja el botó de Vincular dispositiu. 2. El sistema li ensenya quins són els dispositius més propers que hi ha. 3. L'usuari tria el dispositiu que vol. 4. El sistema configura la informació bàsica dintre de l'aplicació. 5. El sistema demana a la placa d'Arduino si s'ha configurat correctament. 6. El sistema carrega la pantalla de mapa.

Propietat	Descripció
ID. Cas d'ús	CU-05
Títol	Desactivació
Precondicions	Ha d'estar la placa d'Arduino activada.
Postcondicions	La placa d'Arduino ja no es pot desactivar fins que no estigui activada de nou.
Escenari principal	<ol style="list-style-type: none"> 1. La placa d'Arduino mira si està connectat al mòbil. 2. La placa troba a prop el dispositiu mòbil. 3. La placa envia una última vegada la posició al servidor. 4. La placa es torna a configurar com al principi.

Propietat	Descripció
ID. Cas d'ús	CU-06
Títol	Configuració Arduino
Precondicions	El mòbil ha d'estar vinculat amb la placa.
Postcondicions	
Escenari principal	<ol style="list-style-type: none"> 1. La placa inicialitza la variable del dispositiu connectat. 2. La placa guarda l'última posició que es té constància. 3. La placa es queda en espera per a una possible alerta.

Propietat	Descripció
ID. Cas d'ús	CU-07
Títol	Enviar posició
Precondicions	El dispositiu mòbil està lluny de la placa.
Postcondicions	
Escenari principal	<ol style="list-style-type: none"> 1. La placa es connecta al servidor. 2. La placa envia al servidor les coordenades per guardar-les. 3. La placa pregunta al servidor si ha guardat correctament les coordenades. 4. La placa espera 10 segons. 5. Tornar al pas 1.

2.2.2 Diagrama de classes

En aquesta part es podrà veure el diagrama que està especialitzat en classes amb una breu explicació de cada una d'ells.

Com es pot veure en la figura [11], podem trobar que l'esquema està dividit en dues parts, la primera seria la placa d'Arduino i la segona seria l'aplicació del dispositiu mòbil. A continuació s'explicarà com actua cada part d'aquestes.

- En la part de la placa es poden veure dues classes principals:

1. Una classe 'main' que és la que s'encarregarà d'inicialitzar la placa quan s'inicia. I poder configurar la placa una vegada que s'hagi vinculat amb el dispositiu mòbil.
2. Una classe 'TinyGPS' que s'encarrega de vincular el xip GSM al dispositiu mòbil i enviar per una part, l'SMS al mòbil i per una altra part les coordenades a la base de dades del servidor.

- En la part de l'aplicació hem de diferenciar dues parts, la primer és la part lògica, que aquesta es compon per:

1. El 'Gestor' de l'aplicació que s'encarregarà de fer totes les crides a les funcions pertinents de les següents classes.
2. Una classe 'Usuario' que serà la que guardarà tota la informació de la persona que tingui l'aplicació
3. La classe 'FetchURL' que s'encarregarà de connectar-se a l'API de Google Maps i descarregar en format JSON tota la informació que ens dóna l'API.
4. Una classe 'DataParser' que s'encarregarà de transformar la informació JSON en informació llegible per a l'aplicació.
5. Una classe 'Data' té l'estructura que pot llegir l'aplicació i s'encarrega de guardar la informació que el retorna el JSON.
6. La interfície 'TaskLoadedCallBack' es la que s'encarrega d'implementar les funcions dintre del Fragment del mapa i de la classe PointsParser, per poder dibuixar el camí de la persona fins a l'animal.

- La segona part de l'aplicació seria la part visual que està dividit per activitats, on cada una d'elles, seria una pantalla, aquestes són:

1. La pantalla 'LoginActivity' és la que s'encarregarà de deixar-te fer l'Inici de sessió dintre de l'aplicació.
2. La pantalla 'RegistroActivity' és la que s'encarregarà de deixar-te fer el registre dintre de l'aplicació.
3. La pantalla 'InicioActivity' és la primera pantalla que veurà l'usuari quan no tingui una sessió una vegada inici l'aplicació.

4. La pantalla 'MainActivity' és la pantalla principal de l'aplicació una vegada la persona estigui loguejat que serà diferent si hi ha algun dispositiu vinculat o no. En el cas que no hi hagi cap dispositiu vinculat es podrà veure el mapa la posició de l'usuari. En el cas que estigui vinculat es veurà la pantalla del mapa amb la posició de l'animal.

Per poder dissenyar l'aplicació es té una idea prèvia del que es vol com a base per poder millorar-la.

Com es pot veure, per una part es té el que seria l'aplicació

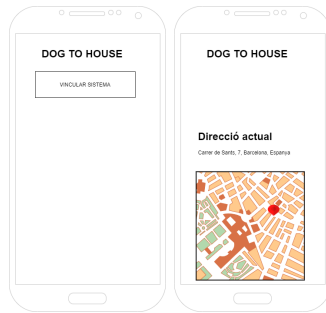


Fig. 3: Interfície base de l'aplicació 'MainActivity'

abans que es vinculés al mòbil i una altra seria quan el mòbil està vinculat i l'animal s'ha escapat i surt el mapa amb la posició exacta d'on és.

2.3 Implementació i resultats

En aquest apartat es podrà veure de forma desglossada les subseccions, (2.3.1,2.3.2,2.3.3,2.3.4) la implementació en un simulador virtual del sistema Dog To House, per poder entendre-ho de la forma més clara possible.

2.3.1 Arduino

La primera part que s'explicarà serà com s'ha implementat l'Arduino en un simulador.

S'ha documentat sobre diferents simuladors que hi ha, entre tots, el que s'ha escollit és Proteus [[8]]. Aquest simulador no és solament d'Arduino, sinó que es pot implementar qualsevol mena de "protoboard". En el cas d'Arduino s'ha d'implementar unes llibreries externes per poder utilitzar dintre d'aquest software, en aquest cas s'han implementat les llibreries de la placa, del Bluetooth i el mòdul que s'encarrega de llegir les coordenades del xip GSM.

En la figura [4] es pot veure tot el Hardware que s'ha implementat diferenciat en dues parts. El primer bloc (Que és la que està a dalt) seria el mòbil amb el seu Bluetooth i el segon bloc (Que és el que aquesta a baix) seria la placa d'Arduino amb els seus mòduls.

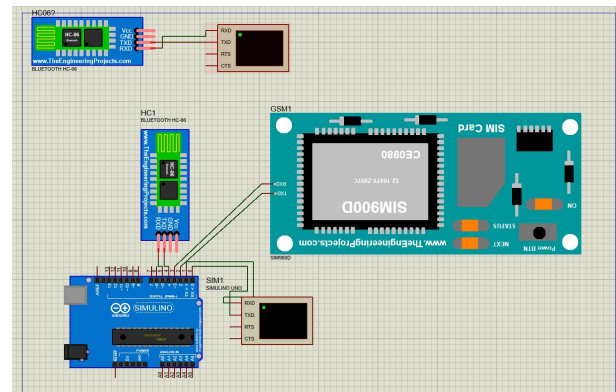


Fig. 4: Sistema Dog To House en Proteus

La part important i on s'ha programat el core principal ha sigut en el bloc d'Arduino, que es pot veure en la figura [5]. El que fa aquesta part és mirar si el Bluetooth està connectat i si aquesta funció retorna que és fals, comença a executar el codi en bucle, agafa les coordenades del xip GSM (placa SIM900) i les envia a la placa d'Arduino que s'encarregarà d'enviar via HTTPS al servidor per poder modificar les dades.

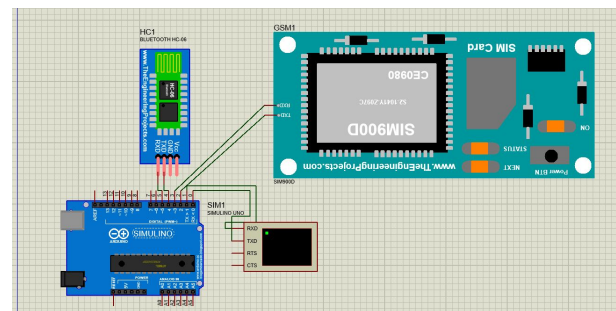


Fig. 5: Arduino en Proteus

Per poder veure el funcionament d'aquesta part s'ha pujat un vídeo de prova on es pot observar com agafa les coordenades (en aquest cas al ser un simulador estan les dades per defecte) i s'envien a l'URL que marca, que en aquest cas és a un servidor local. La prova es pot trobar al vídeo [9].

2.3.2 Servidor

La segona part és el servidor, s'ha implementat un codi que recull la informació de l'URL que arriba des de la placa i amb les llibreries de Firebase [14] s'actualitza la base de dades en temps real, el que permet fer que l'aplicació agafi les coordenades tan aviat com es pugui, oferint a l'usuari una ubicació exacta del seu animal.

El servidor s'ha implementat de forma local, però està preparat per poder fer-ho de forma remota, solament s'hauria de carregar el codi al servidor i canviar l'URL de la placa amb la IP del servidor.

La base de dades que s'utilitza, és la que s'ha comentat, en aquest cas és Realtime Database de Firebase, a continuació es podrà veure una captura de la base de dades.

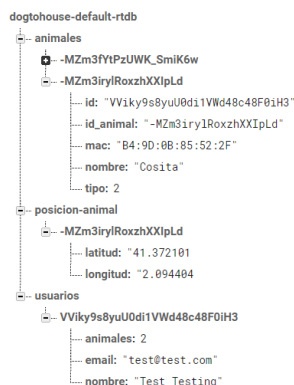


Fig. 6: Realtime Database Firebase

Com es pot veure aquesta base de dades conté la informació imprescindible com id únics per vincular placa i app, informació bàsica de l'usuari i la posició de l'animal amb la qual esta vinculada via la id, com es pot veure en la imatge anterior.

Per acabar amb aquesta part es pot veure una simulació del que passa quan s'activaria l'URL en la placa d'Arduino en el video [10] on es veu que la base de dades s'actualitza en el moment i en la següent part es podrà veure com afecta això en l'aplicació.

2.3.3 Aplicació

En aquesta última part es veurà tot el que té a veure amb la aplicació. Aquesta conté les següents pantalles, les més significatives tindran una imatge d'ajuda:

- Pantalla d'inici de sessió.

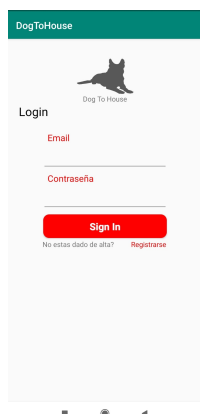


Fig. 7: Inici de sessió de l'aplicació

- Pantalla de Registre.
- Pantalla amb un mapa per veure la posició exacta de l'animal.
- Pantalla per poder veure els dispositius Bluetooth.
- Pantalla per poder crear al teu animal i guardar-ho en la Base de Dades.
- Pantalla amb el teu perfil per poder seleccionar entre animals.

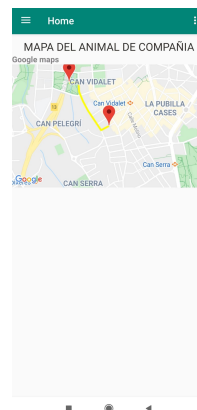


Fig. 8: Mapa de l'aplicació

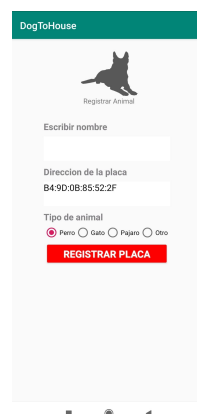


Fig. 9: Vincular animal de l'aplicació

Com s'ha pogut veure en les imatges anteriors es pot fer tot el que s'havia desenvolupat prèviament en la subsecció anterior [2.2]. Per veure una mostra de com funciona l'apartat del mapa es pot trobar en el vídeo [11] on s'actualitza la Base de Dades de Firebase i es veu que la posició s'actualitza de forma automàtica i constant.

2.3.4 Testing i qualitat

En aquest apartat s'explicarà com s'ha obtingut que tot el sistema funcionés correctament i sense errors.

Pel que fa a la part del servidor/placa s'ha testejat que les dades que s'envien mitjançant l'URL que executa la placa acabin guardant-se en la base de dades de Firebase i que la placa retorni els valors per defecte del xip GSM. En aquest cas s'han definit dos testejos amb sortida desitjades, que hi han sigut:

1. Si la placa li entra les dades 43.378 en el paràmetre latitud i 2.091 en el paràmetre longitud, arriba al servidor 43.378 en el paràmetre latitud i 2.091 en el paràmetre longitud.
2. Si en el servidor hi ha les dades 43.378 en el paràmetre latitud i 2.091 en el paràmetre longitud, i actualitzes a les dades 41.378 en el paràmetre latitud i 2.081 en el paràmetre longitud, en la base de dades s'actualitza amb 41.378 en el paràmetre latitud i 2.081 en el paràmetre longitud.

Aquests testejos es poden veure en els vídeos de prova [9] i [10].

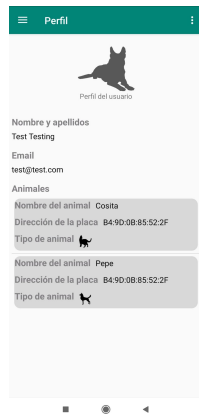


Fig. 10: Perfil de l'aplicació

Per part de l'aplicació s'ha testejat tot el que ha de veure amb la interacció de botons, inputs de text, sortides de pantalla en el mapa, fer el registre amb un usuari i l'inici de sessió amb el mateix usuari. En aquest cas s'han definit els testejos amb sortides desitjades, que hi han sigut:

1. En el registre les entrades han sigut:

- (a) Nom: Test Testing
- (b) Email: test@gmail.com
- (c) Contrasenya: test99
- (d) Número d'animals: 2

Com es pot veure en la figura [6] en l'apartat d'usuari hi han guardades les dades que abans s'han esmentat (cal recordar que en aquesta data base solament es guarden dades essencials, la contrasenya la guarda automàticament Firebase).

2. En el login introduir les dades:

- (a) Email: test@gmail.com
- (b) Contrasenya: test99

La sortida en aquest test seria la pantalla principal amb el mapa, si no s'ha executat, el test ha sigut incorrecte. En el cas que sigui incorrecte l'entrada sigues:

- (a) Email: test@gmail.com
- (b) Contrasenya: test98

La sortida seria la mateixa pantalla i no s'activaria la següent pantalla.

- 3. Si s'ha actualitza la base de dades amb les coordenades x, y. On x és la latitud i y és la longitud, en el mapa es transforma com un punt en la posició x,y. Aquest test es pot veure en el vídeo [11].
- 4. Quan no està el Bluetooth activat avisa per pantalla que no ho tens i en aquest cas et demana que l'activis.
- 5. Quan no hi ha cap dispositiu vinculat et demana que vinculis un dispositiu.
- 6. Quan hi ha un dispositiu vinculat surt en la llista de dispositius vinculats amb el nom i direcció MAC d'aquest dispositiu.

- (a) Nom: Aquaris xPro
- (b) Direcció: "B4:9D:0B:85:52:2F"

En la sortida per pantalla ha de sortir un camp de la llista de dispositius amb les dades del dispositiu vinculat.

- 7. Donar clic a un dispositiu vinculat et porta a la pantalla de crear un animal amb el paràmetre de direcció omplert.
- 8. Omplir la creació d'un animal amb les dades: En aquest cas el dispositiu té les entrades:
 - (a) Nom: Cosita
 - (b) Direcció: "B4:9D:0B:85:52:2F"
 - (c) Tipus: 2

La sortida serà que en la base de dades es guardi aquest animal amb les dades corresponents i que la id que mantingui sigui la del propietari. Aquestes sortides es poden veure en la figura [6] apartat "animals".

- 9. Anar a la pestanya de perfil i veure tota la informació que hi ha guardada en la base de dades, que és la que hi ha en la figura [6].
- 10. Donar-li clic a un animal, que s'actualitzi l'aplicació i que en el mapa et surti la posició d'aquest dispositiu.

Per veure la interacció de les pantalles entre totes elles es poden trobar en els vídeos [11], [12] i [13].

2.4 Estudi de mercat

Per poder fer l'estudi de mercat s'ha buscat diferents proveïdors per poder treure els materials que es necessiten i s'ha sortit al carrer preguntant a un total de 75 persones amb diferents preguntes, per poder saber si aquest sistema es podria treure al mercat o s'hauria de replantejar d'alguna manera.

Pel que es pot veure en la figura [12] els tres proveïdors són:

- 1. Aliexpress, en el cas de producció en massa seria Alibaba perquè aquest fa descompte per quantitat.
- 2. La mateixa web d'Arduino, però aquesta no dona descompte en comprar quantitat.
- 3. Amazon, aquesta té un apartat d'Amazon Business que permet fer compres en gran quantitat.

Es pot observar que Aliexpress és el que menor cost té, però el problema d'aquest seria el temps que tardaria a arribar i després les despeses de l'enviament. L'opció d'Arduino és la més cara, ja que la placa que s'utilitza és un model oficial d'Arduino, no com en els altres dos proveïdors que és una placa Egaloo que funciona igual que l'Arduino (Per aquest sistema).

En la figura [14] es pot veure la quantitat que pagaria la gent per aquest sistema. Es pot observar que la gran quantitat de gent estaria entre 20 €-40 €, però estava majoritàriament sobre els 28 €-32 € el pressupost que tenien. Ja quan es comentava que aquest sistema necessitaria un cost addicional per poder mantenir la xarxa del xip GSM, ja la gent no estava tan interessada a comprar-lo, com es pot veure a la figura. [15].

3 Conclusions

En aquest últim apartat es poden trobar tot el que s'ha tret a partir de tot el projecte en si. La idea principal ha sigut desenvolupar un sistema que fos capaç de poder detectar el teu gos i s'actualitza en un mapa que podria estar en la mà del propietari, dintre del seu mòbil. Com s'ha pogut observar en els apartats (2.1, 2.2, 2.3, 2.3.4), el treball que s'havia decidit fer ha sortit com s'esperava.

El problema a vingut a l'hora de veure si el sistema que s'havia desenvolupat seria fructuós i es podria arribar al mercat. Aquí com s'ha pogut observar en les figures [14] i [15] la gent no estaria disposada a pagar per un manteniment necessari i que potser si es vol fer amb material oficial d'Arduino, no es pot implementar perquè no hi ha benefici.

Com a solució a aquest problema que s'ha trobat, es pot replantejar el sistema com a un sistema educatiu que sirvi per ensenyar a els alumnes a la implementació d'Arduino amb una aplicació Android.

Una altre solució al problema podria ser fer una recerca de un altre tipus de placa que fosin més barates compra-les en quantitats grans i fer tota la instal·lació en la mateixa placa, sense mòduls externs com passa en Arduino. Això pot fer que el producte fos més assequible per al consumidor i poder arribar a un benefici major a l'hora de portar-lo a producció.

Per acabar aquest apartat i el treball s'explicarà els coneixements que s'han adquirit i els que s'han desenvolupat amb aquest Treball Final de Grau.

S'ha adquirit coneixements nous sobre les API de Google, en aquest cas la API coneguda com a Firebase [14] on s'ha pogut desenvolupar tot el sistema d'autenticació i connexió amb base de dades a temps real, un dels punts forts de l'aplicació. Una altre coneixement adquirit ha sigut tot el que té a veure amb Arduino ja sigui des de com s'utilitza fins a la implementació de codi dintre de un simulador. Com a coneixements ampliats, aquests han sigut, per una part el desenvolupament de aplicacions android amb llenguatge de programació JAVA i l'altre part ha sigut el desenvolupament del servidor amb PHP per poder executar la URL que enviava la placa d'Arduino.

Referències

- [1] Agencia de datos - Europa Press "El abandono de animales de compañía, en datos y gráficos," <https://www.epdata.es/datos/abandono-animales-datos-graficos/428#:~:text=Alrededor%20de%20138.400%20perros%20y,y%20consejos%20comarcales%20de%20Espa~na.> 2020.
- [2] Comunidad dfists "Manual de Arduino," <http://dfists.ua.es/~jpomares/arduino/index.html>. 2018.
- [3] Comunidad DescubreArduino "Simulador Arduino, los 7 mejores para PC," <https://descubrearduino.com/simulador-arduino/>. 2019.
- [4] M.Susin - Google Forms "Dog to House: Sistema para encontrar geograficamente al nuestro animal de compañía," <https://forms.gle/TGGg9QsnwdgBjKWB7>. 2021.
- [5] Tractive "Tractive GPS para perros y gatos," <https://tractive.com/es/>. 2021.
- [6] Plant UML "Herramienta de código abierto que utiliza descripciones textuales simples para dibujar diagramas UML," <https://plantuml.com/es/>. 2009.
- [7] Arduino GSM "GSM library," <https://www.arduino.cc/en/Reference/GSM>. 2019.
- [8] Labcenter Electronics "Primeros pasos con la pestaña Diseño PCB," <https://labcenter.s3.amazonaws.com/downloads/pcbTutorialSpanish89.pdf>. 1990.
- [9] M.Susin "Dog to House: Vídeo placa en simulador Proteus," <https://drive.google.com/file/d/14Cak1ZETssOeDCXOzJ2RLh3xgD-ilozG/view?usp=sharing>. 2021.
- [10] M.Susin "Dog to House: Vídeo servidor Firebsae," <https://drive.google.com/file/d/19R1p1F-Y4FJYzFZ-GZc4mtZSsD8VEolB/view?usp=sharing>. 2021.
- [11] M.Susin "Dog to House: Vídeo aplicació mòbil" https://drive.google.com/file/d/1DMKGdKSI29oUOfsruicx1KOHc02S_K-b/view?usp=sharing. 2021.
- [12] M.Susin "Dog to House: Vídeo aplicació mòbil part1," https://drive.google.com/file/d/1fsHrOnayQuqdP1L7T-w0_9h8oeWtZi9F/view?usp=sharing. 2021.
- [13] M.Susin "Dog to House: Vídeo aplicació mòbil part2," <https://drive.google.com/file/d/17SSOBw73oZZmeKMqK1TfvMlubN6Fu6mY/view?usp=sharing>. 2021.
- [14] Firebase - Google "Firebase documentación," <https://firebase.google.com/docs>. 2019.

Apèndix

A.1 Gràfiques disseny

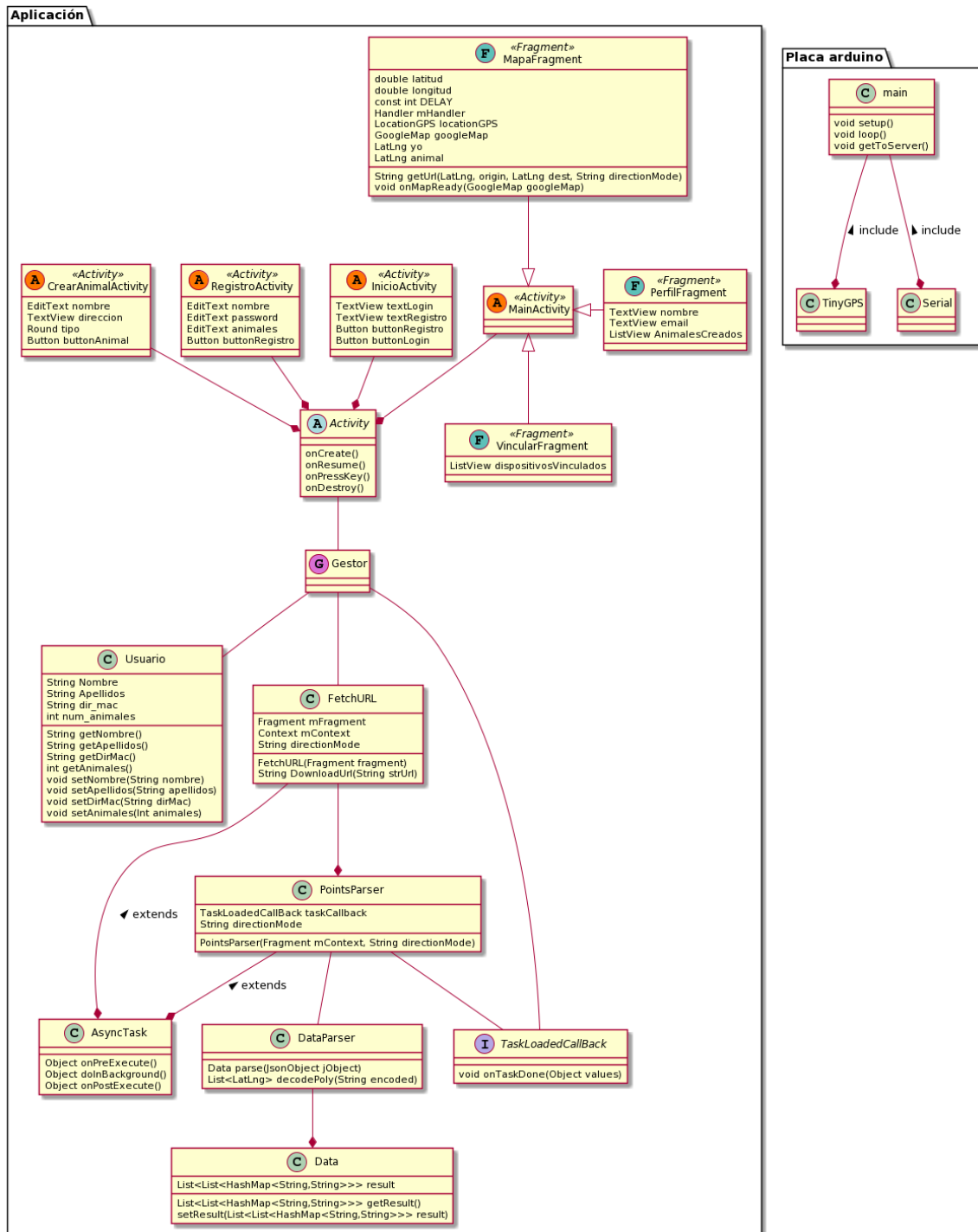


Fig. 11: Diagrama de classes

A.2 Gràfiques d'estudi de Mercat

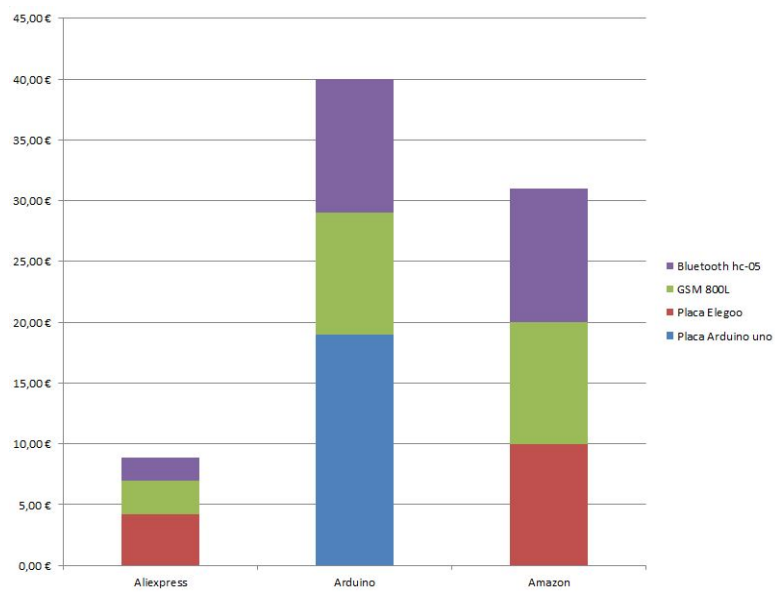


Fig. 12: Pressupost placa (Implementació)

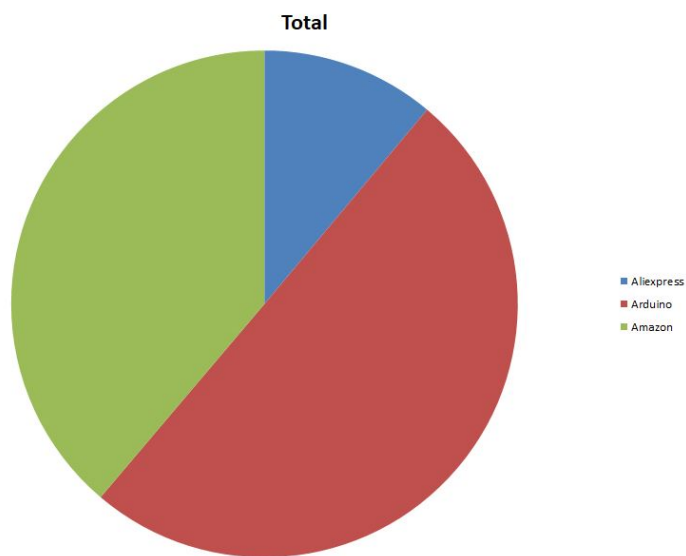


Fig. 13: Pressupost placa (Total)

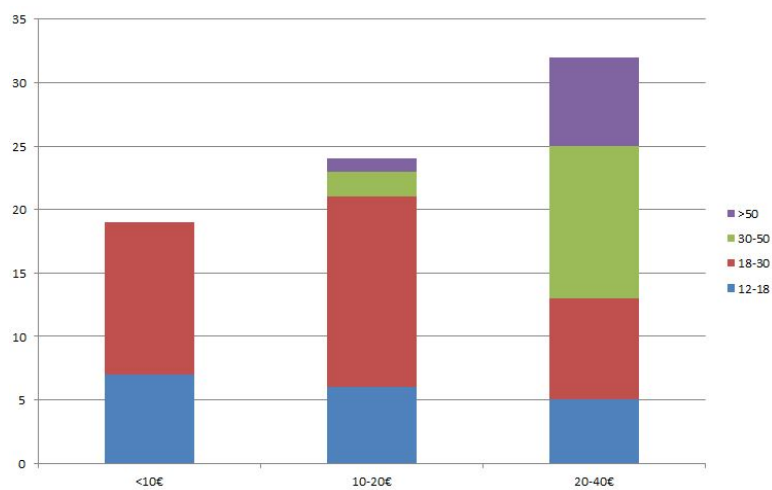


Fig. 14: Preu inicial

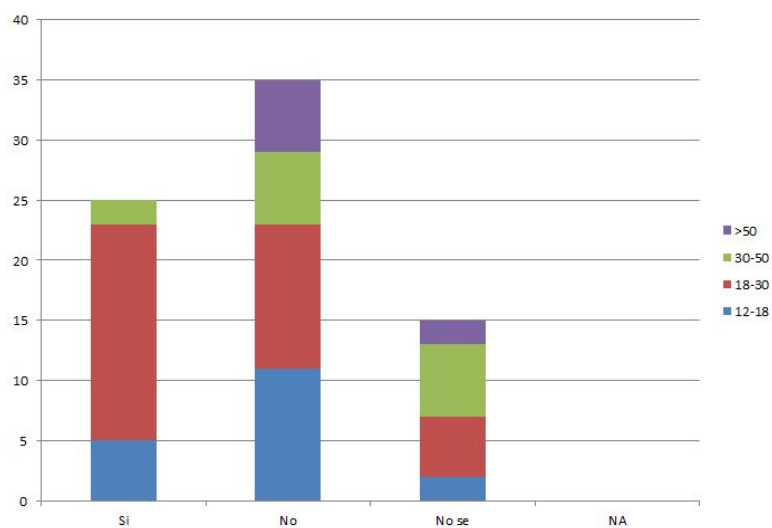


Fig. 15: Preu manteniment